

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior version, and listings, of claims in the application.

Listing of Claims:

1. (currently amended): A process for brokering of locks used on a clustered computer system to control access to Resources of a Cluster by determining whether a Resource on a the Cluster may be locked by a First Node, wherein the Cluster includes the First Node and at least one Peer Node, the process comprising:

communicating a request by a First Node to establish a lock on a Resource accessible through the Cluster;

determining whether the at least one Peer Node on the Cluster holds an active lock on the Resource;

if an active lock on the Resource is not held by any of the at least one Peer Node, approving the lock request; and

if an active lock on the Resource is held by any of the at least one Peer Node further comprising:

determining for each active lock held on the Resource whether the requested lock conflicts with the active lock;

if the requested lock does not conflict with the active lock, approving the lock request; and

if the requested lock conflicts with the active lock, denying the lock request.

2. (original): The process of claim 1, wherein the lock request further comprises:
a lock name;
an intent mode; and
a deny mode, wherein the lock name provides an identification of the Resource.

3. (original): The process of claim 2, wherein the active lock further comprises:
a lock name;

an intent mode; and
a deny mode, wherein the lock name provides an identification of the
Resource.

4. (original): The process of claim 3, wherein the determination of whether an active lock on the Resource is held by any of the at least one Peer Node further comprises:

comparing the lock name of the requested lock with the lock name of each active lock held by each of the at least one Peer Node;

determining that an active lock is held on the Resource if the lock name of the requested lock and the lock name of any active lock held by any of the at least one Peer Node identifies the same Resource; and

determining that an active lock is not held on the Resource if the lock name of the requested lock and the lock name of every active lock held by any of the at least one Peer Node does not identify the same Resource.

5. (original): The process of claim 4, wherein the comparisons of the lock names are accomplished at each of the at least one Peer Node and further comprise examining each entry in a Lock Broker Table.

6. (original): The process of claim 5, wherein each of the at least one Peer Node maintains a separate Lock Broker Table.

7. (original): The process of claim 3, wherein the determination of whether the requested lock conflicts with the active lock further comprises:

comparing the intent mode of the lock request with the deny mode of the active lock; and

comparing the deny mode of the active lock with the intent mode of the lock request;

whereupon failure of either of the comparing steps, the lock request is denied and whereupon passing of both of the comparing steps, the lock request is approved.

8. (original): The process of claim 7, whereupon obtaining approval of the requested lock from each of the at least one Peer Node, the First Node establishes an active lock on the Resource.

9. (original): The process of claim 7, whereupon obtaining a denial of the lock request, the process further comprises;

placing the lock request in a pending state at the Peer Node;

awaiting notification that the active lock which conflicted with the lock request has been released; and

repeating the process identified in claim 1.

10. (original): The process of claim 1, wherein the Resource further comprises:
at least one virtual or real device, accessible through the Cluster, selected from the group consisting of: a data file, a database, a printer, a server, a display monitor, a personal computer, and an element of a personal computer.

11. (original): The process of claim 1, wherein the lock request further comprises a read/write request.

12. (currently amended): A process for implementing a Cluster wide lock broker to control access to Resources on a clustered computer system, comprising:

installing a lock broker daemon on each Node of a Cluster, wherein the Cluster includes at least two Nodes;

establishing a Lock Broker Table associated with each lock broker daemon;

and

determining whether a lock request will be granted by comparing the lock request with each entry in each Lock Broker Table;

whereupon receiving at a First Node a request from a Client to establish a lock on a Resource connected to the Cluster, the lock broker daemon communicates the lock request to each Peer Node on the Cluster; and

whereupon receiving the lock request, each Peer Node determines whether the requested lock conflicts with any active lock already held by a Client associated with the Peer Node by examining the contents of the Lock Broker Table associated with the lock broker daemon for the Peer Node.

13. (original): The process of claim 12, wherein the process is implemented in conjunction with the Cluster management system.

14. (original): The process of claim 12, wherein the process further comprises inserting the lock request as an active lock into the First Node's Lock Broker Table when the lock request is approved by every Peer Node on the Cluster.

15. (original): The process of claim 14, wherein the process further comprises removing the active lock from the First Node's Lock Broker Table when the Client is finished utilizing the Resource.

16. (original): The process of claim 14, wherein the process further comprises deleting the active lock from the First Node's Lock Broker Table when a connection between the First Node and the Resource is disconnected.

17. (original): A computer readable medium containing instructions for determining whether a Client may establish a lock on a Resource accessible through a Cluster, wherein the Client is on a First Node of the Cluster and the Resource is on a Peer Node of the Cluster, by:

- communicating a request by the Client via the First Node to establish a lock on the Resource;

- determining whether at least one Peer Node holds an active lock on the Resource;

- if an active lock on the Resource is not held by any of the at least one Peer Node, approving the lock request; and

- if an active lock on the Resource is held by any of the at least one Peer Node, further comprising:

- determining for each active lock held on the Resource whether the requested lock conflicts with the active lock;

- if the requested lock does not conflict with the active lock, approving the lock request; and

- if the requested lock conflicts with the active lock, denying the lock request.

18. (original): A computer readable medium containing instructions for determining whether a requested lock conflicts with an active lock, wherein each of the

requested lock and the active lock include a lock name, identifying a Resource on a Cluster, an intent mode, and a deny mode; by:

comparing a lock name for the requested lock against the lock name of the active lock;

determining that an active lock is held on a Resource if the lock name of the requested lock and the lock name of the active lock identify the same Resource;

determining that an active lock is not held on the Resource if the lock name of the requested lock and the lock name of the active lock do not identify the same Resource; and

repeating the process for each active lock held by each Peer Node on the Cluster.

19. (original): The computer readable medium of claim 18, wherein each active lock held by a Peer Node is identified in a Lock Broker Table managed by a lock broker daemon on the Peer Node and wherein the lock request is communicated to every Peer Node on the Cluster for the determination of whether an active lock is held on the Resource.

20. (original): The computer readable medium of claim 19, wherein the instructions further include determining whether a conflict exists between the requested lock and an active lock when a determination has been made that an active lock is held on the Resource, by:

comparing an intent mode of the lock request with a deny mode of the active lock; and

comparing a deny mode of the active lock with an intent mode of the lock request;

whereupon failure of either of the comparing steps, the Peer Node associated with the active lock denies the lock request and whereupon passing of both of the comparing steps, the Peer Node associated with the active lock approves the lock request.